

[illegible]

-

- [c8] The computer-implemented builder tool of claim 1, wherein at least one virtual machine of said plurality of virtual machines comprises an interrupt assignment parameter group designator.
- [c9] The computer-implemented builder tool of claim 1, wherein each virtual machine of said plurality of virtual machines comprises an interrupt assignment parameter group designator.
- [c10] The computer-implemented builder tool of claim 8, wherein said parameter detail component comprises a method signature entry field when said interrupt assignment parameter group designator is selected.
- [c11] The computer-implemented builder tool of claim 1, further comprising a new virtual machine creator.
- [c12] The computer-implemented builder tool of claim 11, wherein said new virtual machine creator comprises a routine prompting a user to enter data needed to create a new virtual machine.
- [c13] The computer-implemented builder tool of claim 12, wherein said routine also automatically retrieves data needed to create a new virtual machine.
- [c14] The computer-implemented builder tool of claim 1, further comprising a resource manager.
- [c15] The computer-implemented builder tool of claim 14, wherein said resource manager further comprises a resource determination algorithm.
- [c16] The computer-implemented builder tool of claim 15, wherein said resource manager further comprises a resource modification interface.
- [c17] The computer-implemented builder tool of claim 16, wherein said resource modification interface comprises a resource allocation interface.
- [c18] The computer-implemented builder tool of claim 16, wherein said resource modification interface comprises a resource removal interface.

- [c19] The computer-implemented builder tool of claim 1, further comprising a parameter determination algorithm.
- [c20] The computer-implemented builder tool of claim 19, wherein said parameter determination algorithm comprises a parameter prioritization structure.
- [c21] A method using a computer-implemented builder tool to create a target executable file for a multiple virtual machine environment, comprising the steps of:
- receiving, by the builder tool, compiled source code of a first application;
 - receiving, by the builder tool, compiled source code of a second application;
 - creating, by the builder tool, a first relocatable virtual machine to run the compiled source code of the first application;
 - creating, by the builder tool, a second relocatable virtual machine to run the compiled source code of the second application;
 - determining, by the builder tool, parameters for the multiple virtual machine environment;
 - locating, by a locating tool of the builder tool, said first relocatable virtual machine and said second relocatable virtual machine; and
 - generating a target executable file for the multiple virtual machine environment.
- [c22] The method of claim 21, further comprising the steps of:
- receiving, by the builder tool, compiled source code of a third application;
 - creating, by the builder tool, a third relocatable virtual machine to run the compiled source code of the third application; and
- wherein said locating step said also locates the third relocatable virtual machine.
- [c23] The method of claim 21, further comprising the steps of:
- receiving, by the builder tool, a plurality of additional compiled source codes, each additional compiled source code of said plurality of additional compiled source codes being related to an additional application;
 - creating, by the builder tool, a plurality of additional relocatable virtual

machines, each additional relocatable virtual machine of said plurality of additional relocatable virtual machines being created to run one of the additional compiled source codes of said plurality of additional compiled source codes; and

wherein said locating step said also locates said plurality of additional relocatable virtual machines.

- [c24] The method of claim 21, wherein said determining step further comprises the step of searching for user-defined parameters.
- [c25] The method of claim 24, wherein said determining step further comprises the step of searching for resource defined parameters.
- [c26] The method of claim 25, wherein said determining step further comprises the step of next searching for target hardware configuration defined parameters.
- [c27] The method of claim 26, wherein said determining step further comprises the step of next searching for runtime defined parameters.
- [c28] The method of claim 27, wherein said determining step further comprises the step of next searching for default parameters.
- [c29] The method of claim 21, wherein said compiled source code of a first application comprises a set of first application relocatable objects and an associated set of first application runtime relocatable objects; and wherein said compiled source code of a second application comprises a set of second application relocatable objects and an associated set of second application runtime relocatable objects.
- [c30] The method of claim 21, wherein said generating step further comprises generation of a related list file and a related load script file.
- [c31] The method of claim 24, wherein said step of searching for user-defined parameters comprises searching for an interrupt routine entered as a method signature by a user.

- [c32] The method of claim 24, wherein said step of searching for user-defined parameters comprises searching for user-entered virtual machine build data.
- [c33] The method of step 21, further comprising the step of guiding, by said builder tool, a user to enter data needed to create a new virtual machine.
- [c34] The method of claim 21, further comprising the step of establishing a count of available resources.
- [c35] The method of claim 34, further comprising the step of dynamically updating information on the count of available resources in real time.
- [c36] The method of claim 35, further comprising the step of allocating resources by a user of the builder tool.
- [c37] The method of claim 35, further comprising the step of removing resources by a user of the builder tool.
- [c38] The method of claim 21, further comprising the step of assigning an interrupt routine to a method by entry, by a user of the builder tool, of an identifying method signature.
- [c39] The method of claim 21, further comprising the step of dynamically updating information on the determined parameters in real time.
- [c40] A computer-readable storage medium, comprising a computer-executable code to establish a builder tool for a multiple independent virtual machine environment, said code comprising an algorithm to determine build parameters.
- [c41] The computer-readable storage medium of claim 40, wherein said algorithm to determine build parameters is structured to perform a prioritized search based on how the parameter was defined.
- [c42] The computer-readable storage medium of claim 40, wherein said code further comprises an algorithm to determine available resources.

- [c43] The computer-readable storage medium of claim 42, wherein said algorithm to determine available resources is structured to permit user allocation of an available resource.

- [c44] The computer-readable storage medium of claim 42, wherein said code further comprises an algorithm to dynamically update available resource data throughout the builder tool, in real time, upon determining the available resources.

- [c45] The computer-readable storage medium of claim 40, wherein said code further comprises an algorithm to dynamically update available build parameter data throughout the builder tool, in real time, upon determination of build parameters.

- [c46] The computer-readable storage medium of claim 40, wherein said code further comprises a routine to permit user configuration of hardware pins via a graphical user interface.

- [c47] A computer-readable storage medium, comprising a computer-executable code to establish a builder tool for a multiple independent virtual machine environment, said code comprising an algorithm to determine build parameters, said algorithm to determine build parameters being structured to perform a prioritized search based on how the parameter was defined; said code further comprising an algorithm to determine available resources, said algorithm to determine available resources being structured to permit user allocation of an available resource; said code further comprising an algorithm to dynamically update available resource data throughout the builder tool, in real time, upon determining the available resources; said code further comprising an algorithm to dynamically update available build parameter data throughout the builder tool, in real time, upon determination of build parameters; and said code further comprising a routine to permit user configuration of hardware pins via a graphical user interface.

- [c48] A computer-implemented builder tool to create a target executable file for a

[illegible][illegible][illegible]